Claims

[c1] What is claimed is:

1. An active path extraction method for hardware description language (HDL) code in a computer system, the computer system comprising user input equipment for accepting input from a user, and user output equipment for providing output to the user, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal, the method comprising:
utilizing the user input equipment to obtain at least a start signal from the user, the at least a start signal being an output signal or an input signal;
utilizing the user input equipment to obtain at least a stop signal from the user, the at least a stop signal being an output signal or an input signal;
utilizing the user input equipment to obtain a start time and a stop time;
obtaining simulation results of the electronic circuit from a simulator that utilizes the HDL code, the simulation results having state changes of the output signals and the input signals;
parsing the HDL code to obtain a circuit connection graph between the at least a start signal and the at least a stop signal, the connection graph comprising components, input signals, and output signals that electrically connect the at least a start signal to the at least a stop signal;
utilizing the simulation results and the HDL code to determine which of the components in the circuit connection graph are active components, an active component being any component in the circuit connection graph having an active output signal, an active output signal being any output signal of any component in the circuit connection graph that obtains a state between the start time and the stop time in response to a state change of the at least a start signal according to an execution path of the HDL code; and
utilizing the user output equipment to provide the active components or the active output signals to the user.

[c2] 2. The method of claim 1 in which the user output equipment is utilized to provide both the active components and the active output signals to the user.

[c3] 3. The method of claim 1 wherein the active components are found by iterating

a forward search method, the forward search method comprising:
obtaining an active input signal;
utilizing the connection graph to obtain a connected component, the connected component having the active input signal as an input signal of the connected component;
utilizing the HDL code and the simulation results to obtain the execution path of the connected component at an execution time that is between the start time and the stop time; and
utilizing the execution path to determine if an output signal of the connected component is an active output signal of the connected component, wherein the output signal of the connected component is an active output signal of the connected component only if the output signal of the connected component obtains a state in response to a state change of the active input signal;
wherein the active output signal of the connected component is utilized as an active input signal in a subsequent iteration of the forward search method;
wherein the first iteration of the forward search method utilizes the at least a start signal as the active input signal.

[c4] 4. The method of claim 3 wherein utilizing the execution path to determine if the output signal of the connected component is an active output signal of the connected component comprises:
utilizing the execution path to obtain an equation that generates a value of the output signal of the connected component according to at least an input signal of the connected component, wherein if the active input signal is not within the equation, then the output signal of the connected component is not an active output signal of the connected component; and
parsing the equation to determine if the active input signal within the equation is a required signal for the value of the output signal of the connected component; wherein if the active input signal is a required signal, then the output signal of the connected component is an active output signal of the connected component.

[c5] 5. The method of claim 4 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the

execution time comprises:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and

iteratively substituting simulation results corresponding to a time on or after the execution time into the at least an instruction node according to the at least an instruction node to determine which instruction nodes of the connected component are executed at or after the execution time to obtain the value of the output signal of the connected component, wherein the execution path consists of executed instruction nodes; and

utilizing the execution path to obtain the equation that generates the value of the output signal of the connected component comprises:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the output value of the output signal of the connected component according to at least an input signal of the connected component.

[c6] 6. The method of claim 5 wherein the at least an instruction node of the connected component includes an entry-point node that must be satisfied before other instruction nodes of the connected component are subsequently executed, and the simulation results at the execution time satisfy the entry-point node.

[c7] 7. The method of claim 6 wherein the execution time is a simulation time that is closest to the start time.

[c8] 8. The method of claim 5 wherein the execution time is a simulation time that is closest to a state change of the at least a start signal.

[c9] 9. The method of claim 3 wherein the active input signal has an associated local input time, and the active output signal of the connected component has an associated local output time that is the earliest simulation time on or after the local input time at which the output signal of the connected component obtains a state in response to a state change of the active input signal; wherein the local output time is utilized as a local input time in a subsequent iteration of the

forward search method.

[c10] 10. The method of claim 9 wherein the execution time is a simulation time on or after the local input time.

[c11] 11. The method of claim 9 wherein a differentiated active component is provided to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c12] 12. The method of claim 11 wherein required signals of the active output signal of the differentiated active component are further provided.

[c13] 13. The method of claim 12 wherein values of the required signals at the local output time or the local input time are further provided.

[c14] 14. The method of claim 11 further comprising utilizing the user input equipment to obtain a break point time from the user, and selecting the differentiated active component according to the break point time.

[c15] 15. The method of claim 9 further comprising an animated display of the active components or active signals according to the local input times or the local output times.

[c16] 16. The method of claim 3 further comprising adding the connected component to an active component list if the output signal of the connected component is an active output signal, and adding the output signal of the connected component to an active signal list if the output signal of the connected component is an active output signal.

[c17] 17. The method of claim 1 wherein the active components are found by iterating a backward search method, the backward search method comprising:
obtaining a start-point active output signal;
utilizing the connection graph to obtain a connected component, the connected component having the start-point active output signal as an output signal of the connected component;
utilizing the HDL code and the simulation results to obtain the execution path of the connected component at an execution time that is between the start time

and the stop time; and

utilizing the execution path to determine if an input signal of the connected component is an active input signal of the connected component, wherein the input signal of the connected component is an active input signal of the connected component only if the start-point active output signal obtains a state in response to a state change of the input signal of the connected component; wherein the active input signal of the connected component is utilized as a start-point active output signal in a subsequent iteration of the backward search method;

wherein the first iteration of the backward search method utilizes the at least a stop signal as the start-point active output signal.

[c18] 18. The method of claim 17 wherein the execution time is a simulation time at which the start-point active output signal undergoes a state change.

[c19] 19. The method of claim 17 wherein utilizing the execution path to determine if the input signal of the connected component is an active input signal of the connected component comprises:

utilizing the execution path to obtain an equation that generates the value of the start-point active output signal according to signals within the execution path, wherein if the input signal of the connected component is not within the equation, then the input signal of the connected component is not an active input signal of the connected component; and

parsing the equation to determine if the input signal of the connected component within the equation is a required signal for the value of the start-point active output signal; wherein if the input signal of the connected component is a required signal, then the input signal of the connected component is an active input signal of the connected component.

[c20] 20. The method of claim 19 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time comprises:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical

instruction step of the HDL code; and

utilizing the simulation results corresponding to a time on or before the execution time and the at least an instruction node to back-trace the at least an instruction node to determine which instruction nodes of the connected component are executed on or before the execution time to obtain the value of the start-point active output signal, wherein the execution path consists of all instruction nodes executed to generate the value of the start-point active output signal at the execution time; and

utilizing the execution path to obtain the equation that generates the value of the start-point active output signal comprises:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the value of the start-point active output signal according to signals in the logical statements.

[c21] 21. The method of claim 17 wherein the start-point active output signal has an associated local output time, and the active input signal of the connected component has an associated local input time that is the latest simulation time on or before the local output time at which the active input signal of the connected component undergoes a state change; wherein the local input time is utilized as a local output time in a subsequent iteration of the backward search method.

[c22] 22. The method of claim 21 wherein the execution time is a simulation time on or before the local output time.

[c23] 23. The method of claim 21 wherein a differentiated active component is provided to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c24] 24. The method of claim 23 wherein required signals of the active output signal of the differentiated active component are further provided.

[c25] 25. The method of claim 24 wherein values of the required signals at the local output time or the local input time are further provided.

[c26] 26. The method of claim 23 further comprising utilizing the user input equipment to obtain a break point time from the user, and selecting the differentiated active component according to the break point time.

[c27] 27. The method of claim 21 further comprising an animated display of the active components or active signals according to the local input times or the local output times.

[c28] 28. The method of claim 17 further comprising adding the connected component to an active component list, and adding the input signal of the connected component to an active signal list if the input signal of the connected component is an active input signal of the connected component.

[c29] 29. The method of claim 1 wherein the active components are found by:
iterating a forward search method to obtain a first set of active components, the first iteration of the forward search method utilizing the at least a start signal as an input active signal;
iterating a backward search method to obtain a second set of active components, the first iteration of the backward search method utilizing the at least a stop signal as an output active signal; and
intersecting the first set of active components with the second set of active components to obtain the active components.

[c30] 30. An active forward search method for hardware description language (HDL) code in a computer system, the computer system comprising user input equipment for accepting input from a user, and user output equipment for providing output to the user, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal, the method comprising:
utilizing the user input equipment to obtain at least a start signal from the user, the at least a start signal being an input signal;
utilizing the user input equipment to obtain a start time;
obtaining simulation results of the electronic circuit from a simulator that utilizes the HDL code, the simulation results having state changes of the output signals and the input signals;

parsing the HDL code to obtain a circuit connection graph fanning out from the at least a start signal, the connection graph comprising components, input signals, and output signals;

iterating a forward search method comprising:

obtaining an active input signal;

utilizing the connection graph to obtain a connected component, the connected component having the active input signal as an input signal of the connected component;

utilizing the HDL code and the simulation results to obtain an execution path of the connected component at an execution time that is on or after the start time; and

utilizing the execution path to determine if an output signal of the connected component is an active output signal of the connected component, wherein the output signal of the connected component is an active output signal of the connected component only if the output signal of the connected component obtains a state in response to a state change of the active input signal;

wherein the active output signal of the connected component is utilized as an active input signal in a subsequent iteration of the forward search method, and any component in the connection graph having an active output signal is an active component;

wherein the first iteration of the forward search method utilizes the at least a start signal as the active input signal; and

utilizing the user output equipment to provide the active components or the active output signals to the user.

[c31] 31. The method of claim 30 in which the user output equipment is utilized to provide both the active components and the active output signals to the user.

[c32] 32. The method of claim 30 wherein utilizing the execution path to determine if the output signal of the connected component is an active output signal of the connected component comprises:

utilizing the execution path to obtain an equation that generates a value of the output signal of the connected component according to at least an input signal of the connected component, wherein if the active input signal is not within the

equation, then the output signal of the connected component is not an active output signal of the connected component; and

parsing the equation to determine if the active input signal within the equation is a required signal for the value of the output signal of the connected component; wherein if the active input signal is a required signal, then the output signal of the connected component is an active output signal of the connected component.

[c33] 33. The method of claim 32 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time comprises:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and

iteratively substituting simulation results corresponding to a time on or after the execution time into the at least an instruction node according to the at least an instruction node to determine which instruction nodes of the connected component are executed at or after the execution time to obtain the value of the output signal of the connected component, wherein the execution path consists of all executed instruction nodes; and

utilizing the execution path to obtain the equation that generates the value of the output signal of the connected component comprises:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the value of the output signal of the connected component according to at least an input signal of the connected component.

[c34] 34. The method of claim 33 wherein the at least an instruction node of the connected component includes an entry-point node that must be satisfied before other instruction nodes of the connected component are subsequently executed, and the simulation results at the execution time satisfy the entry-point node.

[c35]

35. The method of claim 34 wherein the execution time is a simulation time

that is closest to the start time.

[c36] 36. The method of claim 34 wherein the execution time is a simulation time that is closest to a state change of the at least a start signal.

[c37] 37. The method of claim 30 wherein the active input signal has an associated local input time, and the active output signal of the connected component has an associated local output time that is the earliest simulation time on or after the local input time at which the output signal of the connected component obtains a state in response to a state change of the active input signal; wherein the local output time is utilized as a local input time in a subsequent iteration of the forward search method.

[c38] 38. The method of claim 37 wherein the execution time is a simulation time on or after the local input time.

[c39] 39. The method of claim 37 wherein a differentiated active component is provided to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c40] 40. The method of claim 39 wherein required signals of the active output signal of the differentiated active component are further provided.

[c41] 41. The method of claim 40 wherein values of the required signals at the local output time or the local input time are further provided.

[c42] 42. The method of claim 39 further comprising utilizing the user input equipment to obtain a break point time from the user, and selecting the differentiated active component according to the break point time.

[c43] 43. The method of claim 37 further comprising an animated display of the active components or active signals according to the local input times or the local output times.

[c44] 44. The method of claim 30 further comprising adding the connected component to an active component list if the output signal of the connected component is an active output signal, and adding the output signal of the

connected component to an active signal list if the output signal of the connected component is an active output signal.

[c45]

45. An active backward search method for hardware description language (HDL) code in a computer system, the computer system comprising user input equipment for accepting input from a user, and user output equipment for providing output to the user, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal, the method comprising:

utilizing the user input equipment to obtain at least a stop signal from the user, the at least a stop signal being an output signal;

utilizing the user input equipment to obtain a stop time;

obtaining simulation results of the electronic circuit from a simulator that utilizes the HDL code, the simulation results having state changes of the output signals and the input signals;

parsing the HDL code to obtain a circuit connection graph fanning into the at least a stop signal, the connection graph comprising components, input signals, and output signals;

iterating a backward search method comprising:

obtaining a start-point active output signal;

utilizing the connection graph to obtain a connected component, the connected component having the start-point active output signal as an output signal of the connected component;

utilizing the HDL code and the simulation results to obtain an execution path of the connected component at an execution time that is on or before the stop time; and

utilizing the execution path to determine if an input signal of the connected component is an active input signal of the connected component, wherein the input signal of the connected component is an active input signal of the connected component only if the start-point active output signal obtains a state in response to a state change of the input signal of the connected component;

wherein the active input signal of the connected component is utilized as a start-point active output signal in a subsequent iteration of the backward

search method;

wherein the first iteration of the backward search method utilizes the at least a stop signal as the start-point active output signal; and

utilizing the user output equipment to provide the active components, the active output signals, or the active input signals to the user.

[c46] 46. The method of claim 45 in which the user output equipment is utilized to provide the active components, the active output signals, and the active input signals to the user.

[c47] 47. The method of claim 45 wherein the execution time is a simulation time at which the start-point active output signal undergoes a state change.

[c48] 48. The method of claim 45 wherein utilizing the execution path to determine if the input signal of the connected component is an active input signal of the connected component comprises:

utilizing the execution path to obtain an equation that generates the value of the start-point active output signal according to signals within the execution path, wherein if the input signal of the connected component is not within the equation, then the input signal of the connected component is not an active input signal of the connected component; and

parsing the equation to determine if the input signal of the connected component within the equation is a required signal for the value of the start-point active output signal; wherein if the input signal of the connected component is a required signal, then the input signal of the connected component is an active input signal of the connected component.

[c49] 49. The method of claim 48 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time comprises:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and

utilizing the simulation results corresponding to a time on or before the execution time and the at least an instruction node to back-trace the at least an

instruction node to determine which instruction nodes of the connected component are executed on or before the execution time to obtain the value of the start-point active output signal, wherein the execution path consists of all instruction nodes executed to generate the value of the start-point active output signal at the execution time; and

utilizing the execution path to obtain the equation that generates the value of the start-point active output signal comprises:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the value of the start-point active output signal according to signals in the logical statements.

[c50] 50. The method of claim 45 wherein the start-point active output signal has an associated local output time, and the active input signal of the connected component has an associated local input time that is the latest simulation time on or before the local output time at which the active input signal of the connected component undergoes a state change; wherein the local input time is utilized as a local output time in a subsequent iteration of the backward search method.

[c51] 51. The method of claim 50 wherein the execution time is a simulation time on or before the local output time.

[c52] 52. The method of claim 50 wherein a differentiated active component is provided to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c53] 53. The method of claim 52 wherein required signals of the active output signal of the differentiated active component are further provided.

[c54] 54. The method of claim 53 wherein values of the required signals at the local output time or the local input time are further provided.

[c55] 55. The method of claim 52 further comprising utilizing the user input equipment to obtain a break point time from the user, and selecting the differentiated active component according to the break point time.

[c56] 56. The method of claim 50 further comprising an animated display of the active components or active signals according to the local input times or the local output times.

[c57] 57. The method of claim 45 further comprising adding the connected component to an active component list, and adding the input signal of the connected component to an active signal list if the input signal of the connected component is an active input signal of the connected component.

[c58] 58. A computer system comprising:
input equipment for obtaining data from a user;
output equipment for providing data to the user;
a central processing unit (CPU) for controlling functionality of the computer system; and
a memory comprising program code that is executable by the CPU to direct operations of the CPU, the program code containing instructions for:
obtaining hardware description language (HDL) code, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal;
utilizing the user input equipment to obtain at least a start signal from the user, the at least a start signal being an output signal or an input signal;
utilizing the user input equipment to obtain at least a stop signal from the user, the at least a stop signal being an output signal or an input signal;
utilizing the user input equipment to obtain a start time and a stop time;
obtaining simulation results generated by a simulator according to the HDL code, the simulation results having state changes of the output signals and the input signals;
parsing the HDL code to obtain a circuit connection graph between the at least a start signal and the at least a stop signal, the connection graph comprising components, input signals, and output signals that electrically connect the at least a start signal to the at least a stop signal;
utilizing the simulation results and the HDL code to determine which of the components in the circuit connection graph are active components, an active component being any component in the circuit connection graph having an

active output signal, an active output signal being any output signal of any component in the circuit connection graph that obtains a state between the start time and the stop time in response to a state change of the at least a start signal according to an execution path of the HDL code; and
utilizing the user output equipment to provide the active components or the active output signals to the user.

[c59] 59. The computer system of claim 58 in which the program code utilizes the user output equipment to provide both the active components and the active output signals to the user.

[c60] 60. The computer system of claim 58 wherein the active components are found by iterating a forward search algorithm in the program code, the forward search algorithm containing instructions for:
obtaining an active input signal;
utilizing the connection graph to obtain a connected component, the connected component having the active input signal as an input signal of the connected component;
utilizing the HDL code and the simulation results to obtain the execution path of the connected component at an execution time that is between the start time and the stop time; and
utilizing the execution path to determine if an output signal of the connected component is an active output signal of the connected component, wherein the output signal of the connected component is an active output signal of the connected component only if the output signal of the connected component obtains a state in response to a state change of the active input signal;
wherein the active output signal of the connected component is utilized as an active input signal in a subsequent iteration of the forward search algorithm;
wherein the first iteration of the forward search algorithm utilizes the at least a start signal as the active input signal.

[c61] 61. The computer system of claim 60 wherein utilizing the execution path to determine if the output signal of the connected component is an active output signal of the connected component contains instructions for:

utilizing the execution path to obtain an equation that generates a value of the output signal of the connected component according to at least an input signal of the connected component, wherein if the active input signal is not within the equation, then the output signal of the connected component is not an active output signal of the connected component; and

parsing the equation to determine if the active input signal within the equation is a required signal for the value of the output signal of the connected component; wherein if the active input signal is a required signal, then the output signal of the connected component is an active output signal of the connected component.

[c62] 62. The computer system of claim 61 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time contains instructions for:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and

iteratively substituting simulation results corresponding to a time on or after the execution time into the at least an instruction node according to the at least an instruction node to determine which instruction nodes of the connected component are executed at or after the execution time to obtain the value of the output signal of the connected component, wherein the execution path consists of executed instruction nodes; and

utilizing the execution path to obtain the equation that generates the value of the output signal of the connected component contains instructions for:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the output value of the output signal of the connected component according to at least an input signal of the connected component.

[c63] 63. The computer system of claim 62 wherein the at least an instruction node of the connected component includes an entry-point node that must be satisfied before other instruction nodes of the connected component are subsequently executed, and the program code contains instructions for parsing the

simulation results to find a value of the execution time at which the entry-point node is satisfied.

[c64] 64. The computer system of claim 62 wherein the execution time is a simulation time that is closest to the start time.

[c65] 65. The computer system of claim 62 wherein the execution time is a simulation time that is closest to a state change of the at least a start signal.

[c66] 66. The computer system of claim 60 wherein the program code further contains instructions for providing the active input signal an associated local input time, and for providing the active output signal of the connected component an associated local output time that is the earliest simulation time on or after the local input time at which the output signal of the connected component obtains a state in response to a state change of the active input signal; wherein the local output time is utilized as a local input time in a subsequent iteration of the forward search algorithm.

[c67] 67. The computer system of claim 66 wherein the program code contains instructions to ensure that the execution time is a simulation time on or after the local input time.

[c68] 68. The computer system of claim 66 wherein the program code contains instructions to provide a differentiated active component to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c69] 69. The computer system of claim 68 wherein the program code contains instructions to provide required signals of the active output signal of the differentiated active component to the user.

[c70] 70. The computer system of claim 69 wherein the program code contains instructions to provide values of the required signals at the local output time or the local input time to the user.

[c71] 71. The computer system of claim 68 further comprising program code to utilize the user input equipment to obtain a break point time from the user, and

program code that selects the differentiated active component according to the break point time.

[c72] 72. The computer system of claim 66 further comprising program code for providing an animated display of the active components or active signals according to the local input times or the local output times.

[c73] 73. The computer system of claim 60 wherein the program code comprises instructions for adding the connected component to an active component list if the output signal of the connected component is an active output signal, and instructions for adding the output signal of the connected component to an active signal list if the output signal of the connected component is an active output signal.

[c74] 74. The computer system of claim 58 wherein the active components are found by iterating a backward search algorithm in the program code, the backward search algorithm containing instructions for:

obtaining a start-point active output signal;

utilizing the connection graph to obtain a connected component, the connected component having the start-point active output signal as an output signal of the connected component;

utilizing the HDL code and the simulation results to obtain the execution path of the connected component at an execution time that is between the start time and the stop time; and

utilizing the execution path to determine if an input signal of the connected component is an active input signal of the connected component, wherein the input signal of the connected component is an active input signal of the connected component only if the start-point active output signal obtains a state in response to a state change of the input signal of the connected component;

wherein the active input signal of the connected component is utilized as a start-point active output signal in a subsequent iteration of the backward search algorithm;

wherein the first iteration of the backward search algorithm utilizes the at least a stop signal as the start-point active output signal.

[c75] 75. The computer system of claim 74 wherein the program code contains instructions to ensure that the execution time is a simulation time at which the start-point active output signal undergoes a state change.

[c76] 76. The computer system of claim 74 wherein utilizing the execution path to determine if the input signal of the connected component is an active input signal of the connected component contains instructions for:
utilizing the execution path to obtain an equation that generates the value of the start-point active output signal according to signals within the execution path, wherein if the input signal of the connected component is not within the equation, then the input signal of the connected component is not an active input signal of the connected component; and
parsing the equation to determine if the input signal of the connected component within the equation is a required signal for the value of the start-point active output signal; wherein if the input signal of the connected component is a required signal, then the input signal of the connected component is an active input signal of the connected component.

[c77] 77. The computer system of claim 76 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time contains instructions for:
parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and
utilizing the simulation results corresponding to a time on or before the execution time and the at least an instruction node to back-trace the at least an instruction node to determine which instruction nodes of the connected component are executed on or before the execution time to obtain the value of the start-point active output signal, wherein the execution path consists of all instruction nodes executed to generate the value of the start-point active output signal at the execution time; and
utilizing the execution path to obtain the equation that generates the value of the start-point active output signal contains instructions for:
linking logical statements derived from each instruction node in the execution

path together by a logical AND operator to obtain a Boolean equation that generates the value of the start-point active output signal according to signals in the logical statements.

[c78] 78. The computer system of claim 74 wherein the program code further contains instructions for providing the start-point active output signal an associated local output time, and for providing the active input signal of the connected component an associated local input time that is the latest simulation time on or before the local output time at which the active input signal of the connected component undergoes a state change; wherein the local input time is utilized as a local output time in a subsequent iteration of the backward search algorithm.

[c79] 79. The computer system of claim 78 wherein the program code contains instructions to ensure that the execution time is a simulation time on or before the local output time.

[c80] 80. The computer system of claim 78 wherein the program code contains instructions to provide a differentiated active component to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c81] 81. The computer system of claim 80 wherein the program code contains instructions for providing required signals of the active output signal of the differentiated active component to the user.

[c82] 82. The computer system of claim 81 wherein the program code contains instructions for providing the values of the required signals at the local output time or the local input time to the user.

[c83] 83. The computer system of claim 80 wherein the program code contains instructions for utilizing the user input equipment to obtain a break point time from the user, and instructions for selecting the differentiated active component according to the break point time.

[c84] 84. The computer system of claim 78 further comprising program code for

providing an animated display of the active components or active signals according to the local input times or the local output times.

[c85] 85. The computer system of claim 74 wherein the program code contains instructions for adding the connected component to an active component list, and adding the input signal of the connected component to an active signal list if the input signal of the connected component is an active input signal of the connected component.

[c86] 86. The computer system of claim 58 wherein the program code contains instructions for:
iterating a forward search algorithm in the program code to obtain a first set of active components, the first iteration of the forward search algorithm utilizing the at least a start signal as an input active signal;
iterating a backward search algorithm in the program code to obtain a second set of active components, the first iteration of the backward search algorithm utilizing the at least a stop signal as an output active signal; and
intersecting the first set of active components with the second set of active components to obtain the active components.

[c87] 87. A computer system comprising:
input equipment for obtaining data from a user;
output equipment for providing data to the user;
a central processing unit (CPU) for controlling functionality of the computer system; and
a memory comprising program code that is executable by the CPU to direct operations of the CPU, the program code containing instructions for:
obtaining hardware description language (HDL) code, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal;
utilizing the user input equipment to obtain at least a start signal from the user, the at least a start signal being an input signal;
utilizing the user input equipment to obtain a start time;
obtaining simulation results generated from a simulator according to the HDL

code, the simulation results having state changes of the output signals and the input signals;

parsing the HDL code to obtain a circuit connection graph fanning out from the at least a start signal, the connection graph comprising components, input signals, and output signals;

iterating a forward search algorithm contained in the program code having instructions for:

obtaining an active input signal;

utilizing the connection graph to obtain a connected component, the connected component having the active input signal as an input signal of the connected component;

utilizing the HDL code and the simulation results to obtain an execution path of the connected component at an execution time that is on or after the start time; and

utilizing the execution path to determine if an output signal of the connected component is an active output signal of the connected component, wherein the output signal of the connected component is an active output signal of the connected component only if the output signal of the connected component obtains a state in response to a state change of the active input signal;

wherein the active output signal of the connected component is utilized as an active input signal in a subsequent iteration of the forward search method, and any component in the connection graph having an active output signal is an active component;

wherein the first iteration of the forward search algorithm utilizes the at least a start signal as the active input signal; and

utilizing the user output equipment to provide the active components or the active output signals to the user.

[c88] 88. The computer system of claim 87 in which the program code contains instruction to utilize the user output equipment to provide both the active components and the active output signals to the user.

[c89] 89. The computer system of claim 87 wherein utilizing the execution path to determine if the output signal of the connected component is an active output

signal of the connected component contains instructions for:
utilizing the execution path to obtain an equation that generates a value of the output signal of the connected component according to at least an input signal of the connected component, wherein if the active input signal is not within the equation, then the output signal of the connected component is not an active output signal of the connected component; and
parsing the equation to determine if the active input signal within the equation is a required signal for the value of the output signal of the connected component; wherein if the active input signal is a required signal, then the output signal of the connected component is an active output signal of the connected component.

[c90] 90. The computer system of claim 89 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time contains instructions for:
parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and
iteratively substituting simulation results corresponding to a time on or after the execution time into the at least an instruction node according to the at least an instruction node to determine which instruction nodes of the connected component are executed at or after the execution time to obtain the value of the output signal of the connected component, wherein the execution path consists of all executed instruction nodes; and
utilizing the execution path to obtain the equation that generates the value of the output signal of the connected component contains instructions for:
linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the value of the output signal of the connected component according to at least an input signal of the connected component.

[c91] 91. The computer system of claim 90 wherein the at least an instruction node of the connected component includes an entry-point node that must be satisfied before other instruction nodes of the connected component are subsequently

executed, and the program code contains instructions for parsing the simulation results to find a value of the execution time at which the entry-point node is satisfied.

[c92] 92. The computer system of claim 91 wherein the program code contains instruction to ensure that the execution time is a simulation time that is closest to the start time.

[c93] 93. The computer system of claim 91 wherein the program code contains instruction to ensure that the execution time is a simulation time that is closest to a state change of the at least a start signal.

[c94] 94. The computer system of claim 87 wherein the program code contains instruction to provide the active input signal an associated local input time, and to provide the active output signal of the connected component an associated local output time that is the earliest simulation time on or after the local input time at which the output signal of the connected component obtains a state in response to a state change of the active input signal; wherein the local output time is utilized as a local input time in a subsequent iteration of the forward search algorithm.

[c95] 95. The computer system of claim 94 wherein the program code contains instructions to ensure that the execution time is a simulation time on or after the local input time.

[c96] 96. The computer system of claim 94 wherein the program code contains instructions to provide a differentiated active component to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c97] 97. The computer system of claim 96 wherein the program code contains instructions to provide required signals of the active output signal of the differentiated active component to the user.

[c98] 98. The computer system of claim 97 wherein the program code contains instructions to provide values of the required signals at the local output time or

the local input time to the user.

[c99] 99. The method of claim 96 wherein the program code contains instructions for utilizing the user input equipment to obtain a break point time from the user, and for selecting the differentiated active component according to the break point time.

[c100] 100. The computer system of claim 94 further comprising program code for providing an animated display of the active components or active signals according to the local input times or the local output times.

[c101] 101. The computer system of claim 87 wherein the program code contains instructions for adding the connected component to an active component list if the output signal of the connected component is an active output signal, and for adding the output signal of the connected component to an active signal list if the output signal of the connected component is an active output signal.

[c102] 102. A computer system comprising:
input equipment for obtaining data from a user;
output equipment for providing data to the user;
a central processing unit (CPU) for controlling functionality of the computer system; and
a memory comprising program code that is executable by the CPU to direct operations of the CPU, the program code containing instructions for:
obtaining hardware description language (HDL) code, the HDL code being used to represent an electronic circuit having a plurality of components, each component having at least an input signal and an output signal;
utilizing the user input equipment to obtain at least a stop signal from the user, the at least a stop signal being an output signal;
utilizing the user input equipment to obtain a stop time;
obtaining simulation results of the electronic circuit from a simulator that utilizes the HDL code, the simulation results having state changes of the output signals and the input signals;
parsing the HDL code to obtain a circuit connection graph fanning into the at least a stop signal, the connection graph comprising components, input signals,

and output signals;

iterating a backward search algorithm comprising:

obtaining a start-point active output signal;

utilizing the connection graph to obtain a connected component, the connected component having the start-point active output signal as an output signal of the connected component;

utilizing the HDL code and the simulation results to obtain an execution path of the connected component at an execution time that is on or before the stop time; and

utilizing the execution path to determine if an input signal of the connected component is an active input signal of the connected component, wherein the input signal of the connected component is an active input signal of the connected component only if the start-point active output signal obtains a state in response to a state change of the input signal of the connected component;

wherein the active input signal of the connected component is utilized as a start-point active output signal in a subsequent iteration of the backward search algorithm;

wherein the first iteration of the backward search algorithm utilizes the at least a stop signal as the start-point active output signal; and

utilizing the user output equipment to provide the active components, the active output signals, or the active input signals to the user.

[c103] 103. The computer system of claim 102 in which the program code contains instructions to utilize the user output equipment to provide the active components, the active output signals, and the active input signals to the user.

[c104] 104. The computer system of claim 102 wherein the program code contains instructions to insure that the execution time is a simulation time at which the start-point active output signal undergoes a state change.

[c105] 105. The computer system of claim 102 wherein utilizing the execution path to determine if the input signal of the connected component is an active input signal of the connected component contains instructions for:

utilizing the execution path to obtain an equation that generates the value of

the start-point active output signal according to signals within the execution path, wherein if the input signal of the connected component is not within the equation, then the input signal of the connected component is not an active input signal of the connected component; and

parsing the equation to determine if the input signal of the connected component within the equation is a required signal for the value of the start-point active output signal; wherein if the input signal of the connected component is a required signal, then the input signal of the connected component is an active input signal of the connected component.

[c106] 106. The computer system of claim 105 wherein utilizing the HDL code and the simulation results to obtain the execution path of the connected component at the execution time contains instructions for:

parsing the HDL code corresponding to the connected component to obtain at least an instruction node, each instruction node corresponding to a logical instruction step of the HDL code; and

utilizing the simulation results corresponding to a time on or before the execution time and the at least an instruction node to back-trace the at least an instruction node to determine which instruction nodes of the connected component are executed on or before the execution time to obtain the value of the start-point active output signal, wherein the execution path consists of all instruction nodes executed to generate the value of the start-point active output signal at the execution time; and

utilizing the execution path to obtain the equation that generates the value of the start-point active output signal contains instructions for:

linking logical statements derived from each instruction node in the execution path together by a logical AND operator to obtain a Boolean equation that generates the value of the start-point active output signal according to signals in the logical statements.

[c107] 107. The computer system of claim 102 wherein the program code contains instructions for providing the start-point active output signal an associated local output time, and for providing the active input signal of the connected component an associated local input time that is the latest simulation time on

or before the local output time at which the active input signal of the connected component undergoes a state change; wherein the local input time is utilized as a local output time in a subsequent iteration of the backward search method.

[c108] 108. The computer system of claim 107 wherein the program code contains instructions to ensure that the execution time is a simulation time on or before the local output time.

[c109] 109. The computer system of claim 107 wherein the program code contains instructions for providing a differentiated active component to the user in a differentiated manner according to the local input time or the local output time of the differentiated active component.

[c110] 110. The computer system of claim 109 wherein the program code contains instructions for providing required signals of the active output signal of the differentiated active component to the user.

[c111] 111. The computer system of claim 110 wherein the program code contains instructions for providing values of the required signals at the local output time or the local input time to the user.

[c112] 112. The computer system of claim 109 wherein the program code contains instructions for utilizing the user input equipment to obtain a break point time from the user, and for selecting the differentiated active component according to the break point time.

[c113] 113. The computer system of claim 107 further comprising program code for providing an animated display of the active components or active signals according to the local input times or the local output times.

[c114] 114. The computer system of claim 102 wherein the program code contains instructions for adding the connected component to an active component list, and for adding the input signal of the connected component to an active signal list if the input signal of the connected component is an active input signal of the connected component.